AMENDMENTS TO THE SPECIFICATION

IN THE ABSTRACT

Please replace the abstract of the disclosure with the attached new abstract of the disclosure.

IN THE SPECIFICATION

Please amend the paragraph beginning on page 2, line 19, as follows:

--FIG. 3 is a schematic illustration showing a decoding deduction for the Viterbi decoder. In FIG. 3, if the Viterbi decoder 200 continuously receives three signals R1, R2 and R3 and will find out the most possible corresponding signal, it firstly computes (performed by the branch metric unit 202 in FIG. 2) the branch metric with respect to each branch in the trellis diagram. The computation method is usually of computing the square value of the signal difference or the absolute value of the signal difference, for example, ~~BM00-00=[R1-(-1.8)]2~~ BM00-00=[R1-(-1.8)]^2 or | R1-(-1.8) |. The so-called "path metric of a certain path" means the addition result of all branch metrics corresponding to all branches of this path. For example, the path metric of the path from State00 at 1T, State00 at 2T, state01 at 3T, and finally to State11 at 4T, indicated by the bold lines in FIG. 3, is "BM00-00+BM00-01+BM01-11".--

Please amend the heading beginning on page 10, line 20, as follows:

--~~DETAIL~~ DETAILED DESCRIPTION OF THE INVENTION--

Please amend the paragraph beginning on page 10, line 21, as follows:

--FIG. 8 is a flow chart showing the method for the Viterbi decoder of the invention. As shown in FIG. 8, a trellis diagram (as shown in FIG. 3) corresponding to the Viterbi decoder is first created in step S70. In order to overcome the bottleneck for the ACS, the trellis diagram of the Viterbi decoder is re-arranged by the re-arranging method including a transverse arrangement and a longitudinal arrangement. Taking the original trellis diagram shown in FIG. 9A and n=2 as an example, FIG. 9B shows a transverse arrangement performed in the original trellis diagram. The so-called transverse arrangement is to merge the trellises ~~of the n sates into a state~~. Although the bottleneck of ACS still exists, the time of the bottleneck is lengthened from 1T to nT. Thus, the overall speed limitations of the Viterbi decoder can be eased.--

Please amend the paragraph beginning on page 11, line 15, as follows:

--FIG. 10 shows a trellis diagram wherein the number of the channel memory of the binary inputs is 3. As shown in FIG. 10, this channel is quite similar to the commonly used PRML channel. Taking FIG. 11, in which the trellis diagram is shown under the limitation of the RLL (2,10) code, as an example, it is found that the trellis diagram in FIG. 11 is simplified a lot under the limitation of the RLL (2, 10) code in the trellis diagram of FIG. 10. As shown in FIG. 12, which shows the result after the longitudinal arrangement in FIG. ~~11is~~ 11 is performed, it is found that the number of states of each stage only increases to 2 under the limitation of the RLL (2, 10) code and after the longitudinal arrangement of the trellis diagram in FIG. 11. As shown in FIG. 13, _which shows the result after the longitudinal arrangement in FIG. ~~11is~~ 11 is performed

twice, it is found that the trellis diagram does not become too complicated but the bottleneck of the ACS has been greatly broadened under the limitation of the RLL (2, 10) code and after two longitudinal arrangements of the trellis diagram in FIG. 11.--

Please amend the paragraph beginning on page 12, line 3, as follows:

--Then, as shown in step S76 in FIG. 8, with respect to each branch in the trellis diagram, the Viterbi decoder calculates the branch ~~matric~~ metric value (performed by the branch metric unit 202 in FIG. 2). Next, the longitudinal arrangement regards the Viterbi decoder as a finite state machine (FSM). It can be known from the property of the finite state machine that:--

Please amend the paragraph beginning on page 12, line 15, as follows:

--It can be known from FIG. 9C, since the trellis diagram changes its property to the function of "current output=only the function of a nextstate", the bottleneck of the ACS ~~dissapears~~ disappears when finding out the survivor path (at this time, the ~~candicate~~ candidate paths cross at the same next state, so their output signals are the same). Because all the branches connected onto this state have the same signal value (because current output=only the function of the next state, and the state is the same), the old path metrics can be compared to determine values of new path metrics when new path metrics (new path metrics=old path metrics+branch metric) are calculated.--